

# Introduction to Dynamic Web Development

## Introduction

In this Topic we will discuss about the key differences between static and dynamic webpages and their development processes. You will also be introduced to the key components of a dynamic website and their functions. In particular, you will be introduced to web servers, database servers, client-side and server-side scripting languages. Finally, you will revise some of your skills in web development by reproducing and testing some static and dynamic web pages.

## Objectives

On completion of the Topic, you should be able to:

1. Set up a local dynamic web development environment using XAMPP
2. Discuss the key components of a dynamic website
3. Identify common client-side and server-side scripting technologies in use today
4. Understand the key difference between dynamic and static web

## Static web vs. Dynamic web

There are basically two main types of web pages - static and dynamic.

- A static web page is one that being served (by web servers) to the client browser **as is** without any intermediate processing. The page content is mainly written in plain HTML and JavaScript, which directly encodes what to be displayed to the user.
- A dynamic web page is one that has to be pre-processed (at the server side) to produce a displayable content (e.g. html page), which will be forwarded to the client browser for displaying. A dynamic page often contains algorithms and/or data structures for creating displayable content from databases or other information sources. The algorithms are commonly implemented in a scripting language like PHP, ASP and JavaScript (which can be used on both server side and client side).

The concept of dynamic web has recently expanded beyond the basic scope of the traditional www service, which was originally designed to display only static pages. Web servers now can process complex algorithms, which can capture an entire business application. The term **web application** has emerged to describe this. Furthermore, dynamic web pages now can also be implemented in programming languages like Java and C#, which are suitable for complex business applications. These pages have to be compiled into runnable objects, which will be served to the user by special types of servers that can execute these objects. These type of servers are referred to as **application**

**servers** in some situation. Application servers can also serve business applications to the clients in addition to displayable contents.

### Activity 1.1

Identify the differences between a static web page and dynamic web page.

A static web page	A dynamic web page
<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;  &lt;p&gt;This is an example of a static page.&lt;/p&gt;  &lt;p&gt;Click the button to display the current date and time on YOUR system.&lt;/p&gt;  &lt;button onclick="myFunction()"&gt;Show Date&lt;/button&gt;  &lt;p id="demo"&gt;&lt;/p&gt;  &lt;script&gt; function myFunction() {   var d = new Date();   document.getElementById("demo").innerHTML = d; } &lt;/script&gt;  &lt;/body&gt; &lt;/html&gt;</pre>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt; &lt;h1&gt;This is an example of a dynamic web page&lt;/h1&gt; &lt;?php echo "The current date and time on the SERVER is: " .date("l h:i:sa"); ?&gt; &lt;/body&gt; &lt;/html&gt;</pre>

## Components of a Dynamic Website

Web browsers communicate with web servers using the Hypertext Transfer Protocol (HTTP). When you click a link on a web page, submit a form, or run a search, an HTTP request is sent from your browser to the target server. A client targets a specific resource through URL (Uniform Resource Locator). The URL is translated to the IP address of the resource address and protocol (HTTP) is used to retrieve it. The resource (e.g. a webpage or error message) is returned back to the requested client through HTTP response message.

In static websites, web servers serve only static pages. The pages are retrieved from the file system and send directly to the clients. Typical components of a static website is given in Fig 1.1.

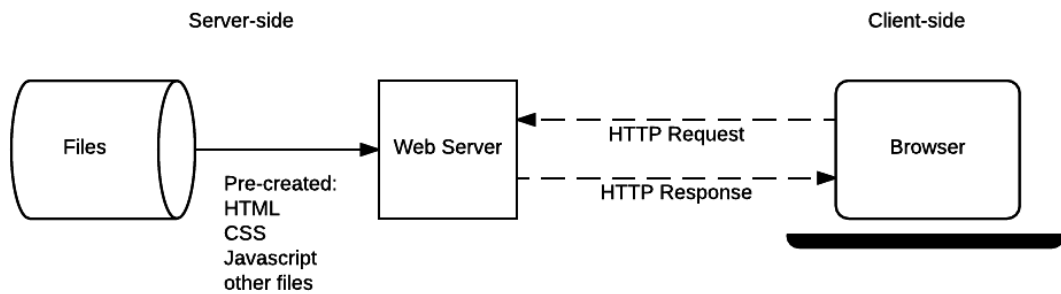


Fig 1.1 Components of a static website

In dynamic websites, web servers serve both static and dynamic pages. Static pages are served in the same way they are served in static websites. Requests for dynamic pages are redirected to the web application processing module where the corresponding pages/objects are retrieved from the file system and processed. If the database access is required, the application module will contact the database server as directed by the algorithms being processed. The output results in displayable format e.g. HTML is returned back to the web server before being forwarded to the client browser for displaying. Fig 1.2 shows typical components of a dynamic website.

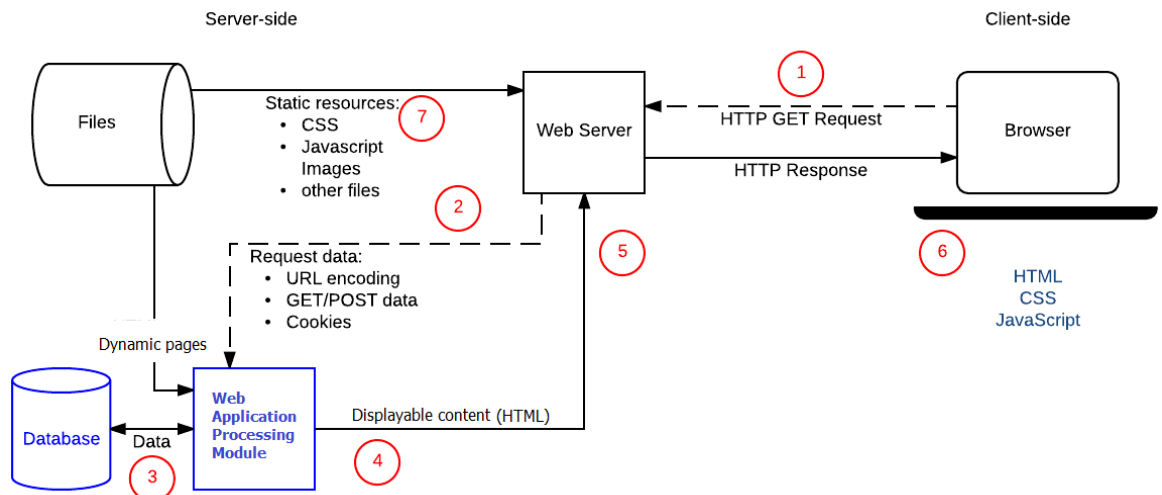


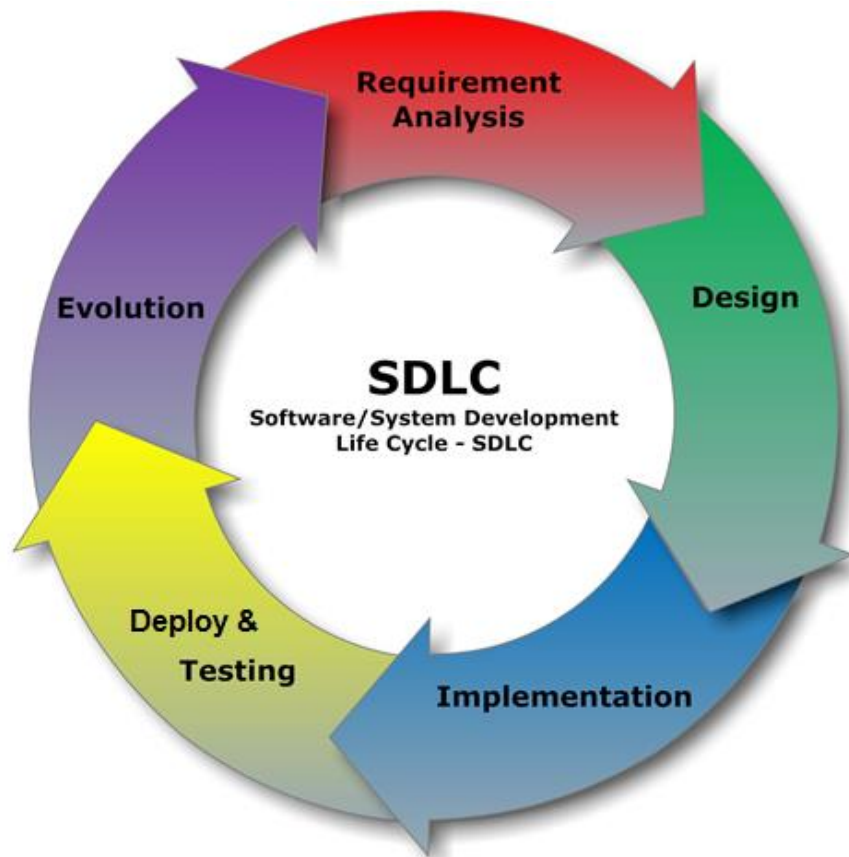
Fig 1.2 Components of a dynamic website

(Reference source: [https://developer.mozilla.org/en-US/docs/Learn/Server-side/First\\_steps/Introduction](https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Introduction))

## Dynamic Web Development

### Development Process

The development process of a web application follows the standard software development process, which is an **iterative** cycle of the following steps:



- **Requirement Analysis:** gather application functional and non-functional requirements.
- **Design:** perform high-level system design and low-level module design
- **Implementation:** implement the design

- **Deployment and Testing:** deploy and test the application at different levels
- **Evolution:** a new development cycle begins with new requirements and modification

### Activity 1.2

Identify a functional and a non-functional requirement of the **moviezone** application (provided in the assignment-1 description) and post it to the unit forum under week 1 topic.

### Development Environment

Dynamic web development begins with setting up a working environment for deploying and testing dynamic web pages. Depending on the server-side technology different software packages will be needed. In this unit, you will use PHP as the server-side technology. So in general you will need:

- A functional web server with PHP support.
- A working SQL database server
- Web programming tools

Although there are different web and database servers, we recommend the use of the Apache webserver, MySQL database and PHP5 in this unit. Using a different web server, database or a different version of PHP may cause compatibility issues.

In this unit you will be provided with an account on **infotech.scu.edu.au** machine, which has a functioning Apache web server, MySQL database and uses PHP5. However, the **infotech** server is **not suitable** for the development process because you do not have local access to the server. As a result, you will have to upload the development code to the servers via FTP for testing, which is time consuming and requires the Internet access. Therefore, it is recommended that you install a local development environment on your own development computer.

There are different ways to setup a suitable development environment depending on your level of knowledge and skills. If you are a beginner, it is strongly recommended that you use XAMPP software package. XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database (an implementation of MySQL database), and interpreters for scripts written in the PHP and Perl programming languages. It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes.

### Development Tools

Dynamic web development require:

- Client-side development tools: in general all html tools are suitable for the client side development. If you prefer to work directly with html code then **notepad++** is an excellent html editor. If you want some fancy WYSIWYG editors then Aptana Studio, Bluefish and CoffeeCup can be a good choice.
- Server-side development tools: server-side development tools are often technology specific. However, some multipurpose editors such as **notepad++** can be used to write server-side code in different programming languages.
- Integrated Development Environment (IDE): due to the complexity of the development process, IDE's are often used as an all-in-one solution for editing, compiling and testing software applications. NetBeans, Eclipse and Ms. Visual Studio are examples of IDE's. Today's IDE often supports different type of application development and technologies. All the mentioned above IDE are perfect tools for web application development.

### Activity 1.3

Download and install XAMPP in your local PC. Use the following website to help you with the setup: <http://webdevzoom.com/install-xampp-on-windows-complete-guide/>. Note that Apache web server runs on TCP port 80 and 443; and MySQL server runs on port 3306 by default. If you cannot start the servers, it is likely that some other applications e.g. Skype may already use these ports. In this case, you will need to change the default ports or stop the other applications. After the installation, start the Apache and MySQL servers and make sure they are all running without any issues (the server names are in green). Troubleshoot the issues if necessary.

You will need to configure and test XAMPP before you can use it for the development. The following configuration is required as the minimum:

- Configure the Apache web server **document root** directory to the desired folder. The document root is a folder on the server file system (in this case is your PC file system) where the web server will search for any requested web pages and related resources.
- Create a database access account on the MySQL server using phpMyAdmin tool.

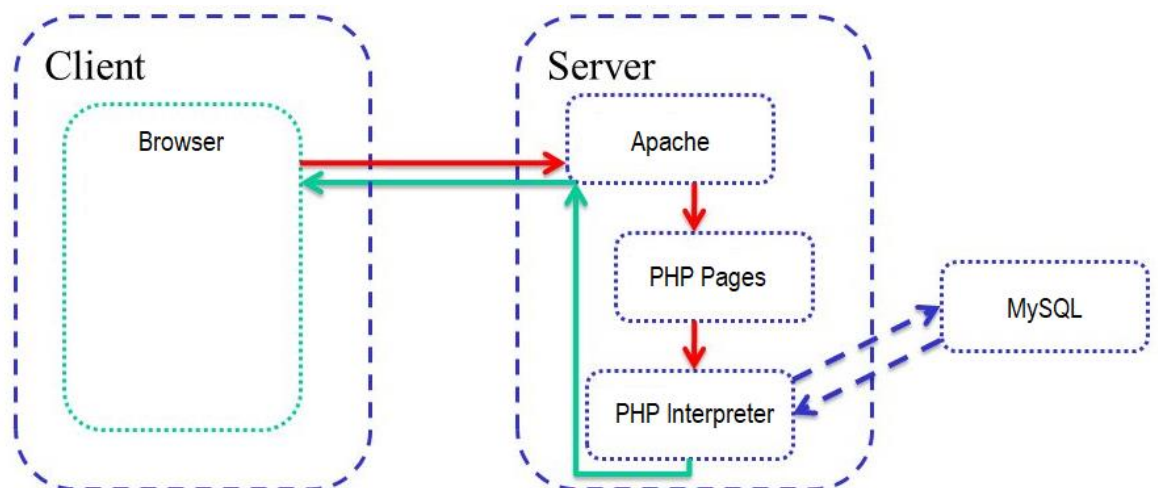
Follow the video tutorials under **week-1** topic to configure the Apache and MySQL server.

In this unit we will use **notepad++** for both client-side and server-side development. So in this activity, you will download and install notepad++ on you PC and configure it for the development.

- Download and install notepad++ from: <https://notepad-plus-plus.org/download/>
- Install notepad++ plugins. Use the following link: <http://webdevzoom.com/13-best-notepad-plugins-fast-development/> to find out about the plugins, which are potentially useful for the web application development. Note that new notepad++ does not have plugin manager pre-installed so you may want to install the plugin manager manually from this link: <https://github.com/bruderstein/nppPluginManager/releases>. Alternatively, you can install any plugins manually by copying them directly to the nodepad++ plugins folder.

### Activity 1.4

The following diagram shows a dynamic website architecture with its components. Use Fig 1.2 as a reference, redraw the architecture and place the components in the right place. For each of the component, find out the installed version of the corresponding software on your XAMPP setup.



## Client-side and server-side scripting technologies

### Client-side scripting technologies

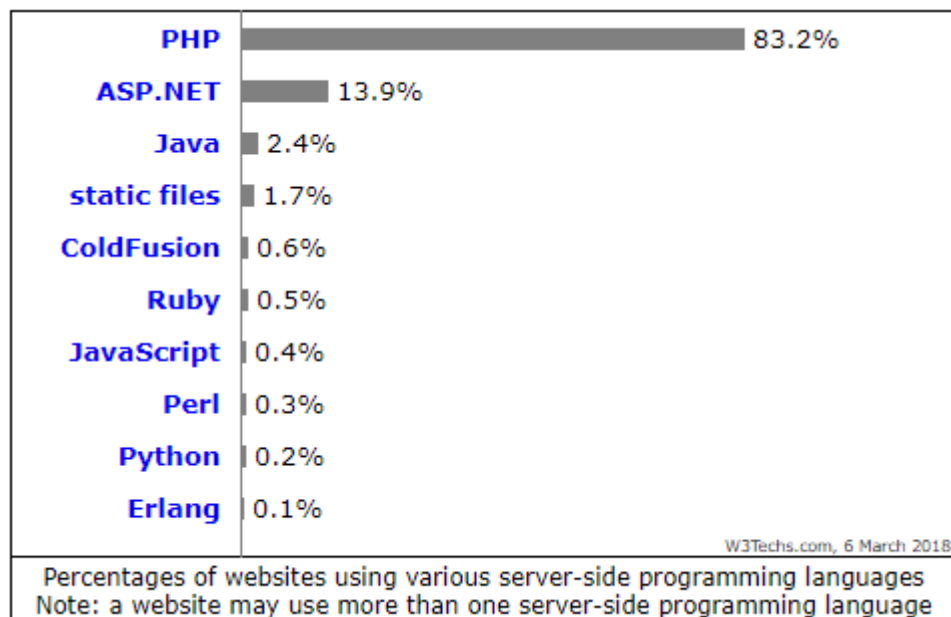
Client-side scripts are embedded in a webpage HTML code, which is hosted on the server in a language that is "understood" by the client browser. The browser temporarily downloads that code, and then executes it. Since scripts are executed at the client-side, they are called client-side scripts.

Although there were a number of client-side scripting technologies/languages e.g. Flash, Silverlight and JavaScript, the only technology remained widely supported today is JavaScript. Together with HTML and CSS, JavaScript is the core of the World Wide Web content engineering.

### Server-side scripting technologies

Server-side scripts are used to implement algorithms in dynamic web pages. The scripts are hosted and executed on the servers (not clients), hence they are called server-side scripts. Only displayable contents resulted from the script execution are returned to the client for displaying.

Unlike on the client-side, there are considerable numbers of server-side scripting technologies that continue to be supported. Examples of server side scripting languages are PHP, ASP, Perl, Ruby, Cold Fusion and Python (there are more).



According to w3Tech.com, PHP is the most popular scripting technology in use today and this is the reason for using PHP in Web Development II.

### Activity 1.5

Reproduce the static and dynamic web pages given in Activity 1.1 and test them on your XAMPP. Discuss the key differences between static web and dynamic web. Post your answer and the screen shots on the unit forum under week 1 topic to show your testing results.

## Assignment Advice

We should start doing Assignment-1 from this week. You will identify all functional and non-functional requirements of the movie-zone application as described in the assignment task.

## Summary

In this topic, you have been introduced to the dynamic web technologies and components of a dynamic website. You now should be able to compare and contrast dynamic websites and static websites in several aspects such as development process, service architecture and working mechanisms. You should also be able to identify the key components of a dynamic website and discuss their functions in processing and serving dynamic web pages. Finally, you should be familiar with the key client-side and server-side scripting technologies i.e. JavaScript and PHP.

In the next topic - Topic 2 - you will learn even more about dynamic web development. In particular, we will explore web applications, their components, design architectures and development environment. You will have chance to install and test some interesting web applications.

## References

MDN Web Docs. 2018. Introduction to the server side - Learn web development | MDN. [ONLINE] Available at: [https://developer.mozilla.org/en-US/docs/Learn/Server-side/First\\_steps/Introduction](https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Introduction). [Accessed 06 March 2018].